東北大学機関リポジトリ
TOUR
Tohoku University Repository

グラフクラスに基づくシュタイナー木遷移問題に関する研究

著者	水田 遥河
雑誌名	東北大学電通談話会記録
巻	86
号	1
ページ	262-263
発行年	2017-08
URL	http://hdl.handle.net/10097/00121462

修士学位論文要約（平成29年３月）

グラフクラスに基づくシュタイナー木遷移問題に関する研究

水田 遥河

指導教員: 周 暁，学位論文指導教員: 伊藤 健洋

The Steiner Tree Reconfiguration Problem on Specific Graph Classes

Haruka MIZUTA

Supervisor: Xiao ZHOU, Research Advisor: Takehiro ITO

For a graph and a vertex subset, called a terminal set, a Steiner tree is a subtree of the graph which contains all terminals. We study a reconfiguration problem for Steiner trees, defined as follows: Given a graph, a terminal set, and two Steiner trees, the Steiner tree reconfiguration problem is to determine whether there exists a sequence of Steiner trees that transforms a given Steiner tree into another one by exchanging a single edge at a time. In this thesis, we analyze the computational complexity of Steiner tree reconfiguration from the viewpoint of graph classes, and give an interesting picture of the boundary between intractability and polynomial-time solvability.

1. Introduction

The Steiner tree problem on graphs is one of the most well-known NP-complete problems[2]. For an unweighted graph G and a vertex subset $S \subseteq V(G)$, a *Steiner tree* for S is a subtree of G which includes all vertices in S; each vertex in S is called a *terminal*. For example, Fig. 1 illustrates four Steiner trees of the same graph G for the same terminal set S. Given an unweighted graph G, a terminal set $S \subseteq V(G)$, and an integer $k \geq 0$, the Steiner tree (search) problem is to determine whether there exists a Steiner tree T of G for S such that T consists of at most k edges. This problem is known to be NP-complete even for planar graphs[2].

The concept of Steiner trees has several applications such as network routing and VLSI design. In the network routing problem, a graph represents a computer network such that each terminal corresponds to a user or a server, each non-terminal to a router, and each edge to a communication link. Then, we wish to find a routing which connects all users and severs to provide the service; thus, a Steiner tree of the graph represents such a routing.

However, the network routing problem could be considered in more “dynamic” situations: In order to temporarily remove routers for maintenance, we sometimes need to change the current routing (i.e., Steiner tree) into another one. To minimize disruption, this transformation needs to be done by switching communication links one by one, while keeping the connectivity among all users and servers to provide the service even during the transformation.

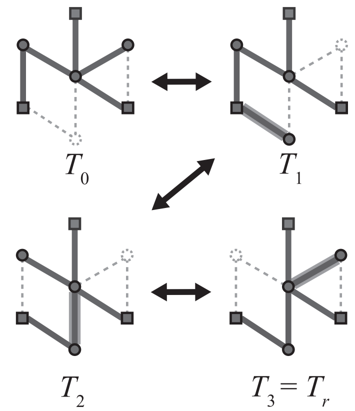

Fig. 1. A sequence $\langle T_0, T_1, T_2, T_3 \rangle$ of Steiner trees, where the terminals are depicted by squares, non-terminals by circles, the edges in Steiner trees by thick lines.

In this thesis, we thus study the following problem: Suppose that we are given two Steiner trees of a graph G for a terminal set $S \subseteq V(G)$ (e.g., the

© 2017 東北大学電気通信研究所

upper-left and lower-right ones in Fig. 1), and we are asked whether we can transform one into the other via Steiner trees for S such that each Steiner tree in the transformation can be obtained from the previous one by exchanging a single edge, that is, two consecutive Steiner trees T and T' satisfy both $|E(T) \setminus E(T')| = 1$ and $|E(T') \setminus E(T)| = 1$. We call this decision problem the STEINER TREE RECONFIGURATION problem. For the particular instance of Fig. 1, the answer is yes as illustrated in the figure.

2. Known results

Ito et al.[3] studied the SPANNING TREE RECONFIGURATION problem, which can be seen as STEINER TREE RECONFIGURATION when restricted to the case where all vertices in a given graph are terminals. They showed that any instance of SPANNING TREE RECONFIGURATION is a yes-instance, that is, there always exists a desired transformation between two spanning trees in any graph.

3. Our results

In this thesis, we study the computational complexity of STEINER TREE RECONFIGURATION from the viewpoint of graph classes. In particular, we deal with graph classes called split graphs, cographs, and interval graphs[1]. Figure 2 summarizes our results, together with the inclusion relationship between graph classes. (A preliminary version of our results appeared in the proceedings of IWOCA 2016[4].)

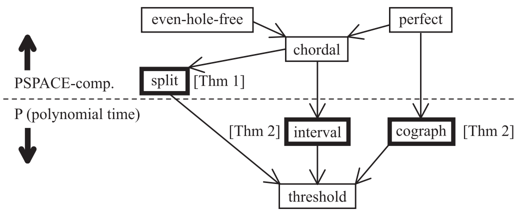

Fig. 2. Our results, where each arrow represents the inclusion relationship between graph classes: $A \to B$ represents that a graph class B is properly included in a graph class A[1].

We first give the following computational hardness of the problem.

Theorem 1 STEINER TREE RECONFIGURATION *is PSPACE-complete for split graphs.*

As a proof of Theorem 1, we gave a polynomial-time reduction from VERTEX COVER RECONFIGURATION, which is known to be PSPACE-complete[3], to our problem for split graphs

We then show that STEINER TREE RECONFIGURATION is solvable in polynomial time for some graph classes.

Theorem 2 STEINER TREE RECONFIGURATION *can be solved in polynomial time for cographs, and for interval graphs.*

As a proof of Theorem 2, we constructed such polynomial-time algorithms. Indeed, we proposed a general scheme to handle the reconfigurability of Steiner trees. We proved that the scheme can yield polynomial-time algorithms for cographs, and for interval graphs.

4. Conclusion

In this thesis, we have shown that STEINER TREE RECONFIGURATION is PSPACE-complete even for split graphs (and hence for chordal graphs and for perfect graphs), while solvable in polynomial time for interval graphs and for cographs. Thus, we have clarified an interesting boundary on the graph classes lying between intractability and tractability, because the structure of split graphs (resp., chordal graphs) can be seen as a star-like (resp., tree-like) structure of cliques, while that of interval graphs can be seen as a path-like structure of cliques.

References

1) Brandstädt, A., Le, V.B., Spinrad, J.P.: Graph Classes: A Survey. SIAM (1999)
2) Garey, M.R., Johnson, D.S.: Computers and Intractability: A Guide to the Theory of NP-Completeness. Freeman, San Francisco (1979)
3) Ito, T., Demaine, E.D., Harvey, N.J.A., Papadimitriou, C.H., Sideri, M., Uehara, R., Uno, Y.: On the complexity of reconfiguration problems. Theoretical Computer Science 412, pp. 1054–1065 (2011)
4) Mizuta, H., Ito, T., and Zhou, X.: Reconfiguration of Steiner trees in an unweighted graph. Proc. of IWOCA 2016, LNCS 9843, pp. 163–175 (2016)